

Ejercicio computacional: Para hacer este ejercicio vamos a usar los datos orbitales del fichero **IGS12651.SP3**. Leedlos como siempre usando `read_sp3()`.

a) (55%) La influencia de la posición de los satélites se mide con el PDoP (Position Dilution of Precision). Valores altos del PDoP indican que la posición calculada puede tener mayores errores. Para calcular el PDoP partimos de la matriz H y calculamos Q (matriz 4×4) como: $Q = (H^T \cdot H)^{-1}$ (en MATLAB $H^T = H'$). El PDoP se calcula a partir de la diagonal de la matriz Q :

$$PDoP = \sqrt{Q_{11} + Q_{22} + Q_{33}} \quad (1)$$

Para la posición **pos = [4853000; -326000; 4113000]** vamos a calcular su PDOP a lo largo de todo un día usando los datos del campo `sp.XYZ` ($3 \times 96 \times 29$) con las posiciones XYZ de los 29 satélites cada cuarto de hora. En un script:

- Reservar un vector `pdop` de tamaño 1×96 para guardar los resultados.
- Hacer bucle desde $k=1$ a 96 (los 96 tiempos del fichero). En cada paso:
 - 1) Extraer la posición de todos los satélites en el k -ésimo instante de tiempo haciendo `XYZ=squeeze(sp.XYZ(:,k,:))`. `squeeze()` compacta los datos extraídos en una matriz de 3 coordenadas \times 28 satélites.
 - 2) Usar la función `elaz` para calcular la elevación de los satélites y usarla para determinar qué satélites son visibles.
 - 3) Quedaros SOLO con la posición XYZ de los satélites visibles y para dichos satélites, calcular la matriz H usando `get_HR()`.
 - 4) Calcular la matriz $Q = (H^T \cdot H)^{-1}$ y a partir de ella el PDoP según la fórmula (1). Guardarlo en la correspondiente casilla del vector `pdop`.

Una vez terminado el bucle presentar en una gráfica la evolución del PDoP en función de la hora del día. Adjuntad código script y la gráfica.

¿Cuál es el mínimo y el máximo PDoP a lo largo del día?

¿Qué hora sería la mejor para tomar observaciones?

b) (25%) Extraer el error de reloj correspondiente al satélite con PRN 17 a lo largo de todo el día y hacer una gráfica de dicho error en milisegundos (1seg = 1000 msec) Adjuntad código y la gráfica. A partir de ella determinar el error de reloj al inicio y al final del día en microsegundos (1 msec = 1000 μ sec)

La especificación del sistema GPS obliga al Centro de Control a volver a poner en hora el reloj de un satélite si su error alcanza 1 milisegundo. ¿Cuántos días faltan para que eso suceda para este satélite?

La deriva de un reloj mide lo que se adelanta o atrasa. Para relojes de cuarzo es del orden de unas pocas partes por millón (ppm). Un reloj con una deriva de 1 ppm se atrasa (o adelanta) 1 segundo cada millón de segundos. O, lo que es lo mismo, 1 microsegundo (μ seg) cada segundo. Determinar a partir de los resultados anteriores la deriva del reloj de este satélite en ppm.

scribble

96


c) (20%) Durante ese día un GPS tomó las siguientes observaciones a las 12h del mediodía (según su reloj).

```
satélites observados = [ 3    11    14    15    19    22    31];  
distancias a satélites =[21734073.8; 21687083.4; 23585628.9; 23986390.1;  
                        20574251.7; 23360156.6; 20293231.8];
```

Obtener tow (time of week) correspondiente a esa hora.

Usando este tow y las medidas a los satélites usar vuestra rutina `get_pos` (versión básica) para obtener la posición del GPS. Usar como punto de partida la posición **pos = [4853000; -326000; 4113000]** y volcad la posición en cada paso de la iteración con 1 decimal, obteniendo un volcado similar a éste (con diferentes números):

```
Iter 0: 4853000.000 -326000.000 4113000.000  
Iter 1: 4849833.480 -335021.921 4116007.770  
...
```

Adjuntad vuestro volcado. ¿Cuántas iteraciones necesitáis?

Volver a resolver usando ahora solo los 4 primeros satélites y dar la distancia entre las dos posiciones obtenidas.